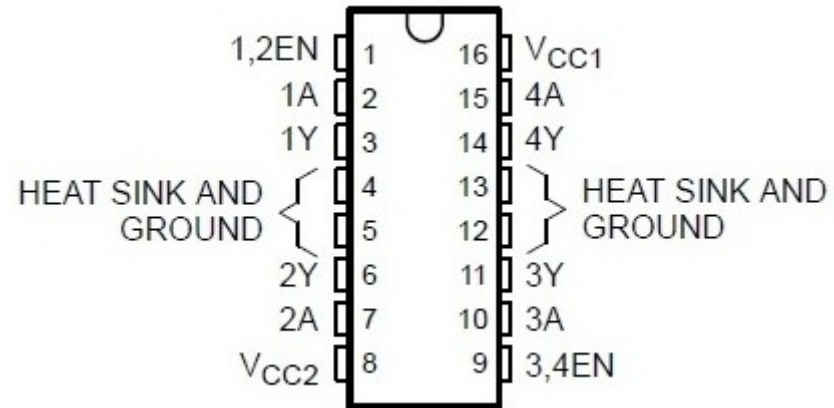


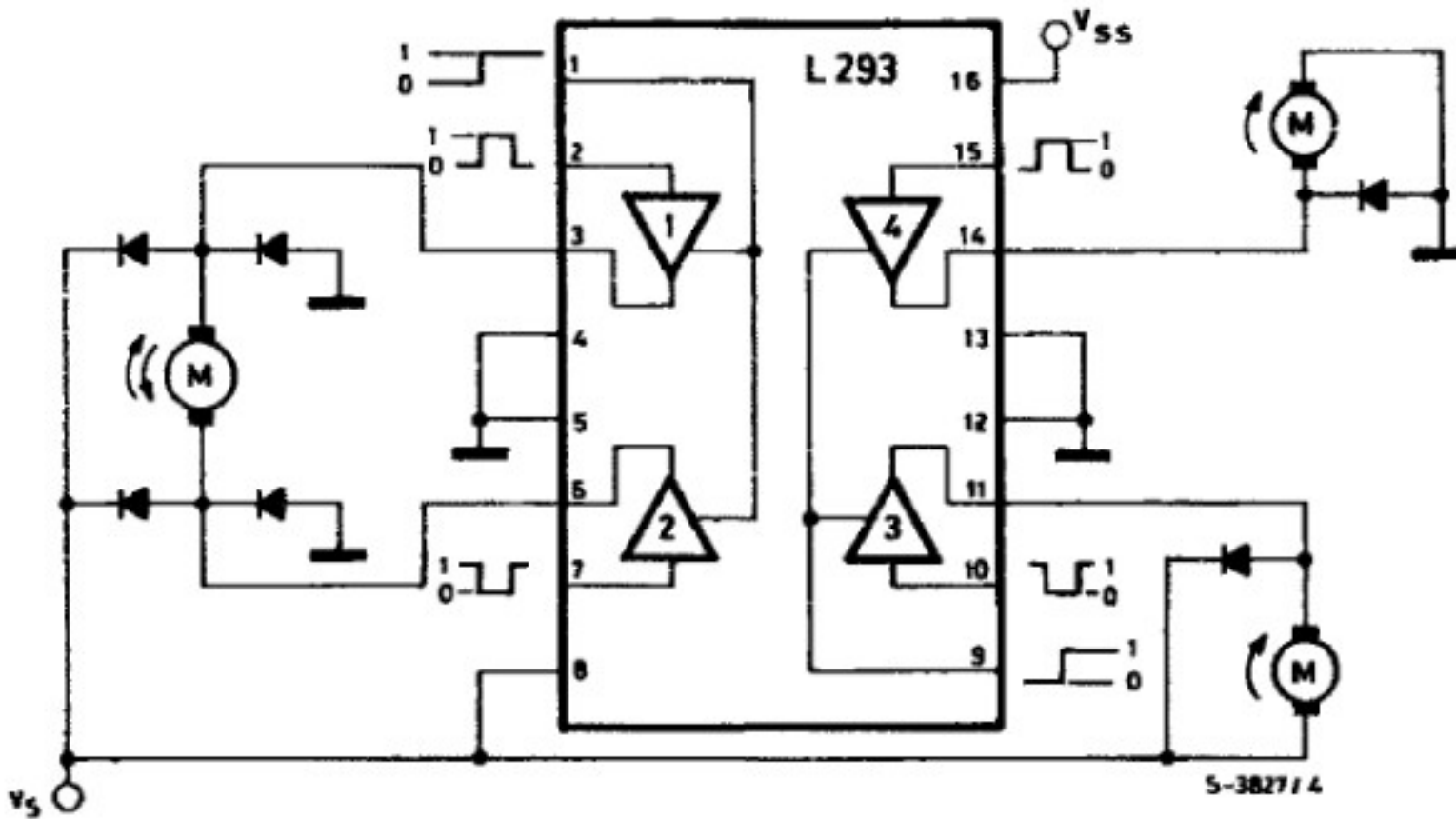
# Controle de motores

L293



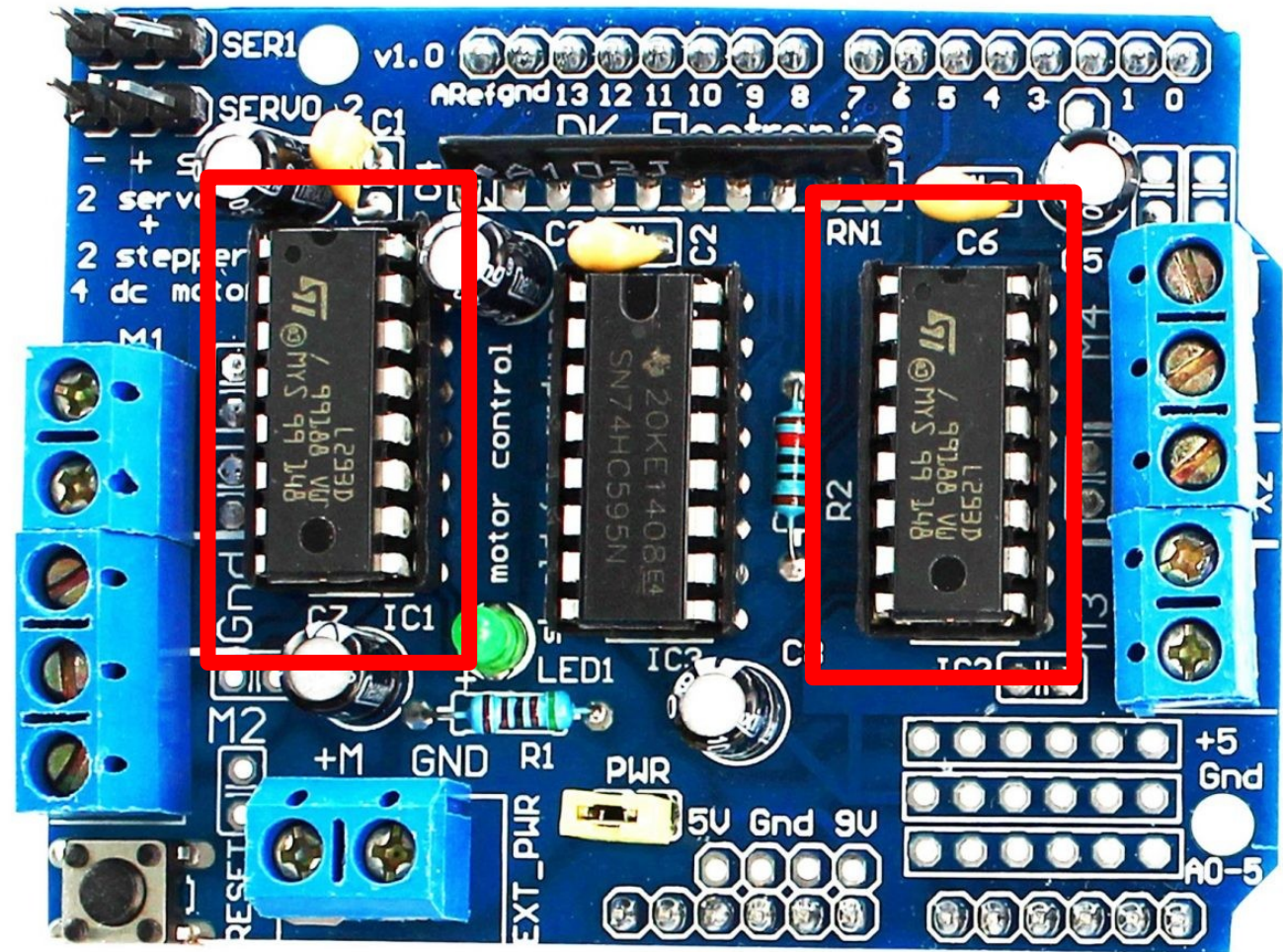


DIP16 - L293B



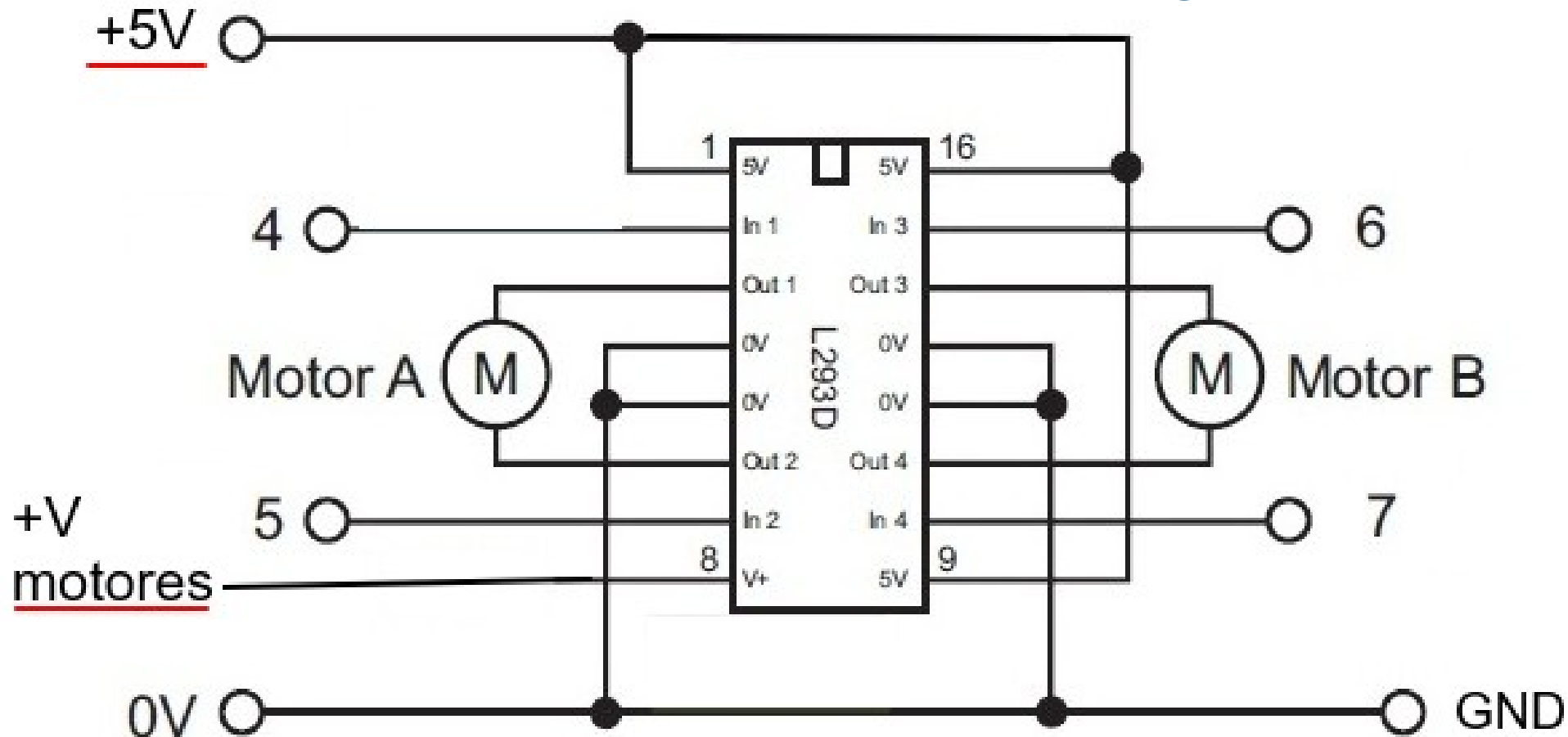
O circuito interno do L293B pode controlar 4 bobinas, sendo até 4 motores girando em sentido único, ou 2 motores girando em dois sentidos ou 1 motor de passo de 4 bobinas.

Nosso módulo possui 2 circuitos L293B, colocados em soquetes (podem ser retirados para testes ou trocados em caso de problemas).

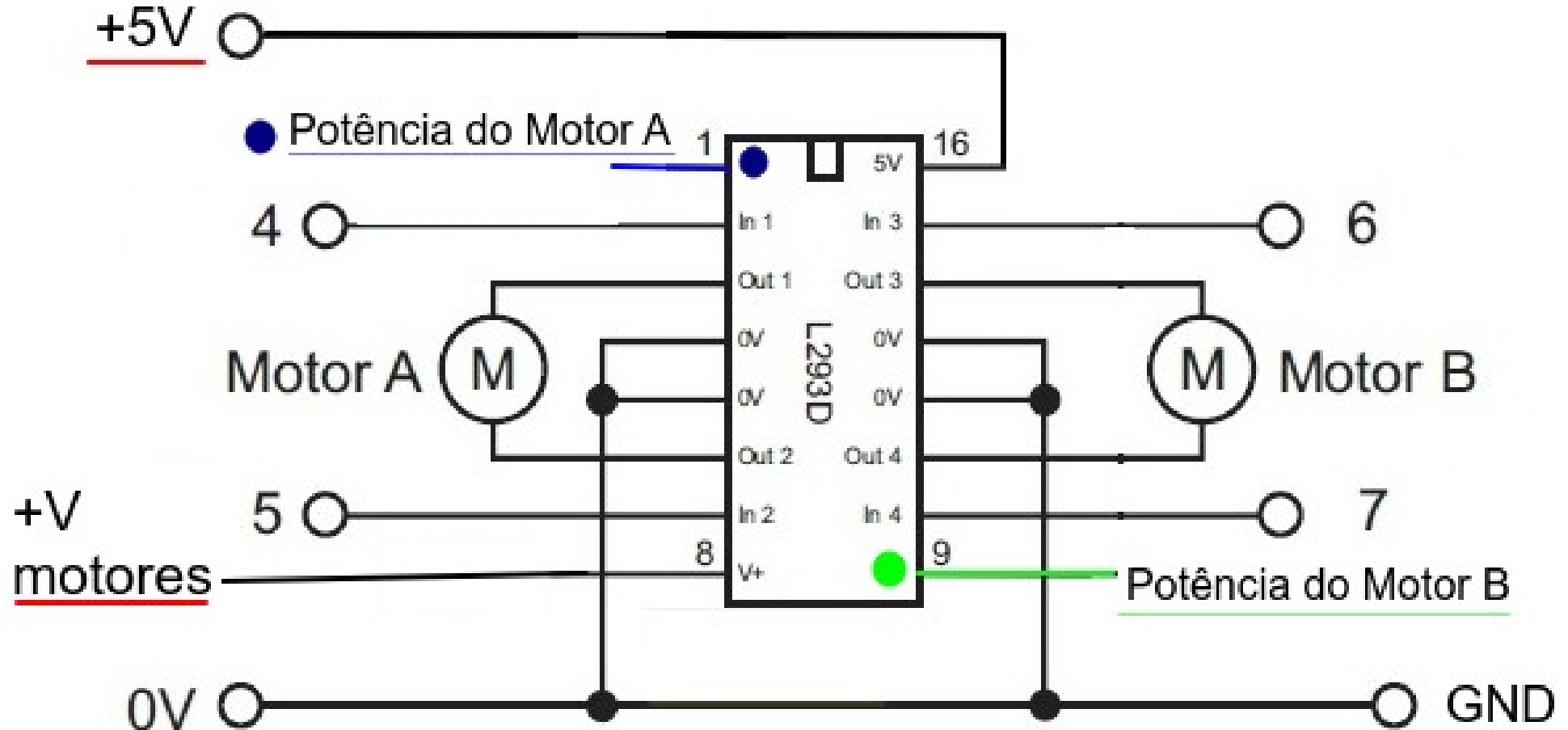


# Circuito básico

# Velocidade fixa – motores sempre com a máxima energia



# Velocidade variável – motores com a energia ajustável (*PWM*)



# Modo de controle do sentido de rotação e/ ou não rotação

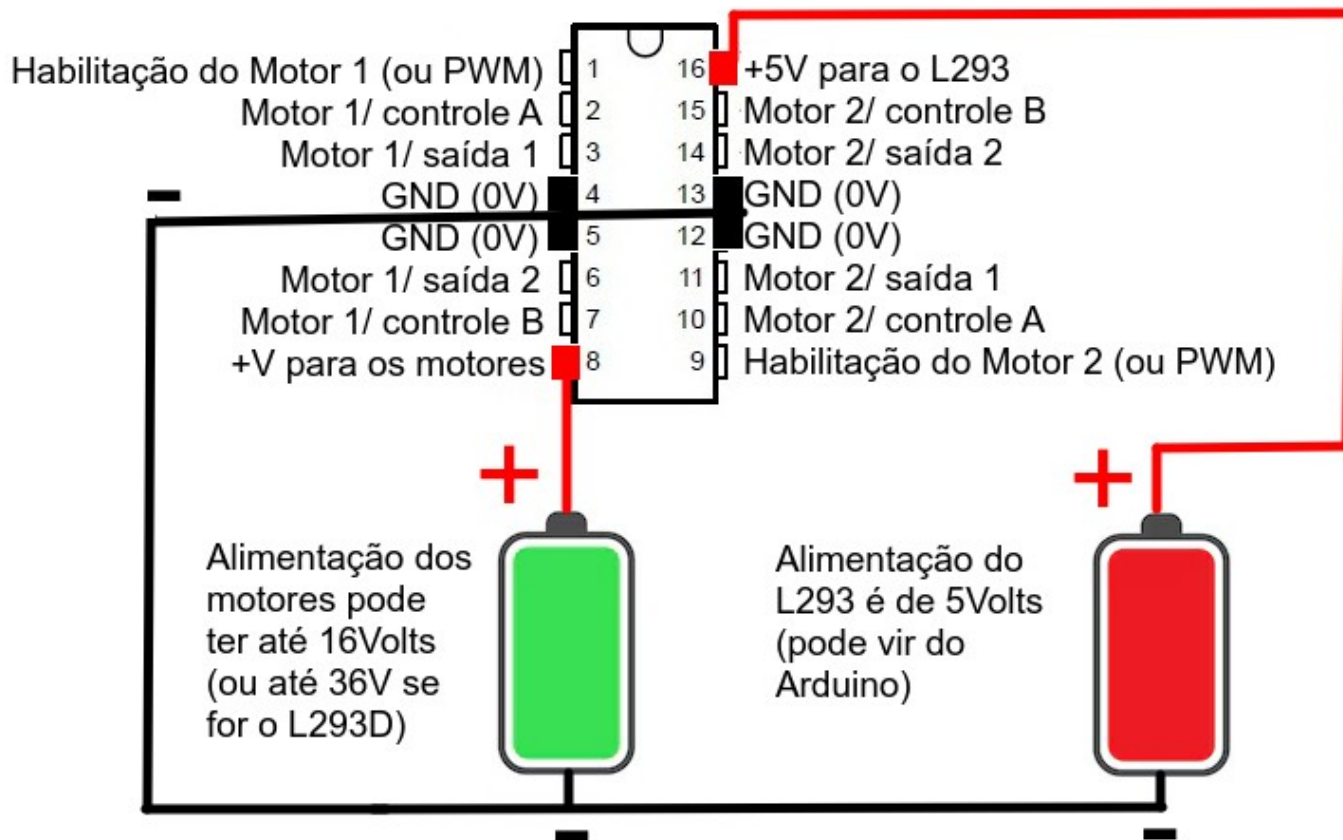
Pino	Nível	Motor 'A' ligado entre 3 e 6
1	LOW	Motor em 3 e 6 desabilitado
1	HIGH (ou PWM)	Motor em 3 e 6 dependendo dos sinais dos pinos 2 e 7. Energia pode ser controlada por PWM
Pinos 2 e 7 dependem do pino 1		
2	LOW	3 e 6 desligado
7	LOW	
2	LOW	3 e 6 ligado em um sentido
7	HIGH	
2	HIGH	3 e 6 ligado em sentido inverso
7	LOW	
2	HIGH	Inválido
7	HIGH	



# Modo de controle do sentido de rotação e/ ou não rotação – motor ‘B’ entre pinos 11 e 14

Pino	Nível	Motor ‘B’ ligado entre 11 e 14
9	LOW	Motor em 11 e 14 desabilitado
9	HIGH (ou PWM)	Motor em 11 e 14 dependendo dos sinais dos pinos 10 e 15. Energia pode ser controlada por PWM
Pinos 11 e 14 dependem do pino 9		
10	LOW	11 e 14 desligado
15	LOW	
10	LOW	11 e 14 ligado em um sentido
15	HIGH	
10	HIGH	11 e 14 ligado em sentido inverso
15	LOW	
10	HIGH	Inválido
15	HIGH	

# ENERGIA – a alimentação dos motores possui um pino em separado e pode ser maior do que 5V



# Experimentos

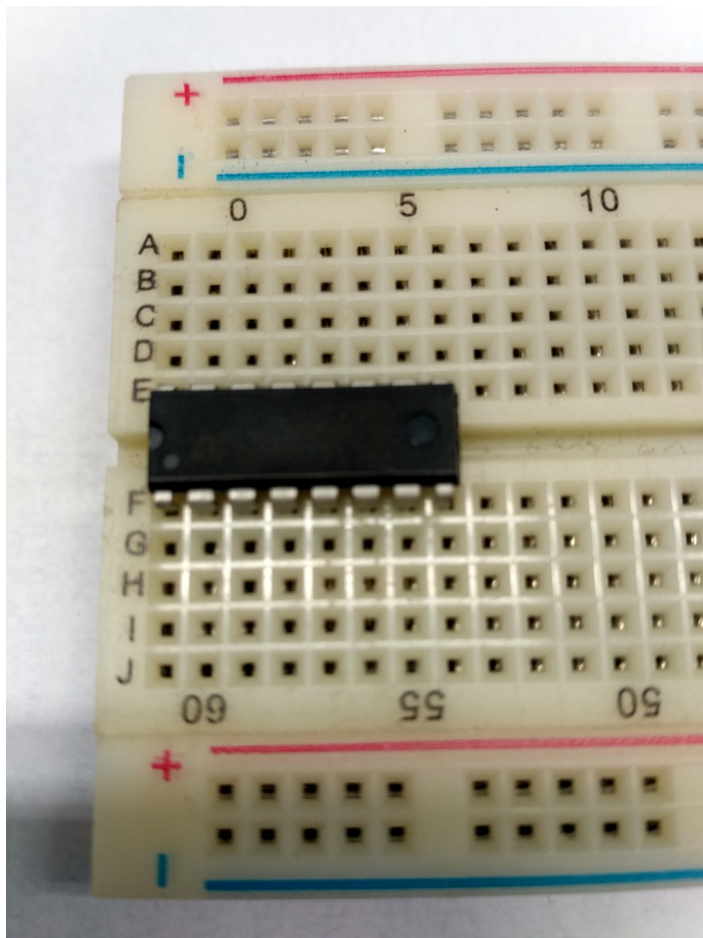
Retire com cuidado um dos circuitos L293 do  
módulo



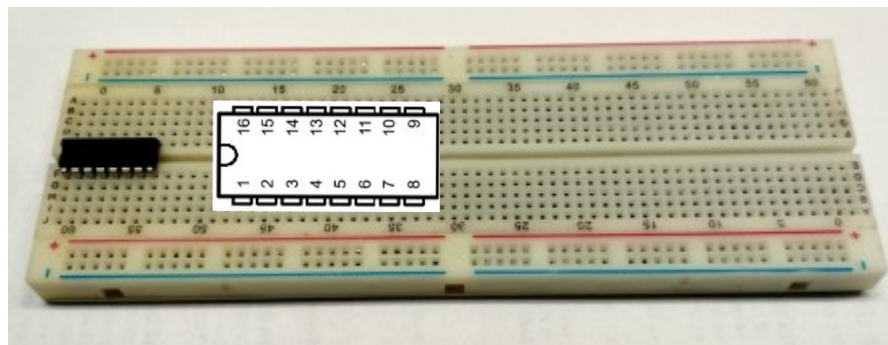
Pino '1'



Retire com cuidado um dos circuitos L293 do módulo



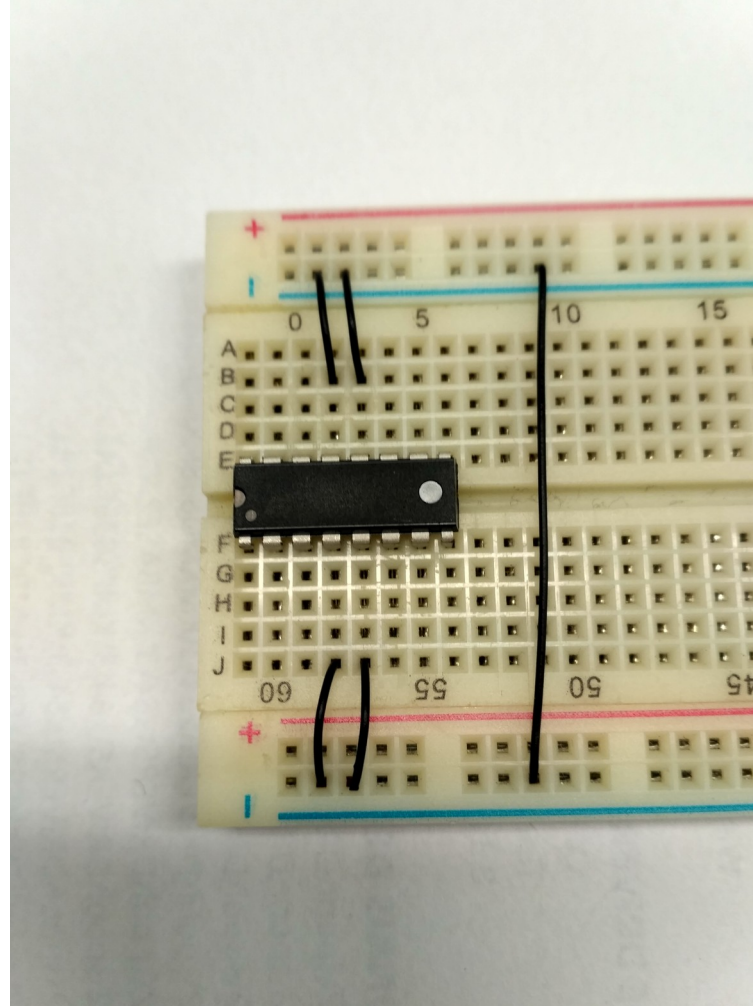
Coloque-o na *breadboard*



# Ligações – conexão GND

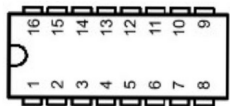


Pinos GND  
4, 5, 12, 13



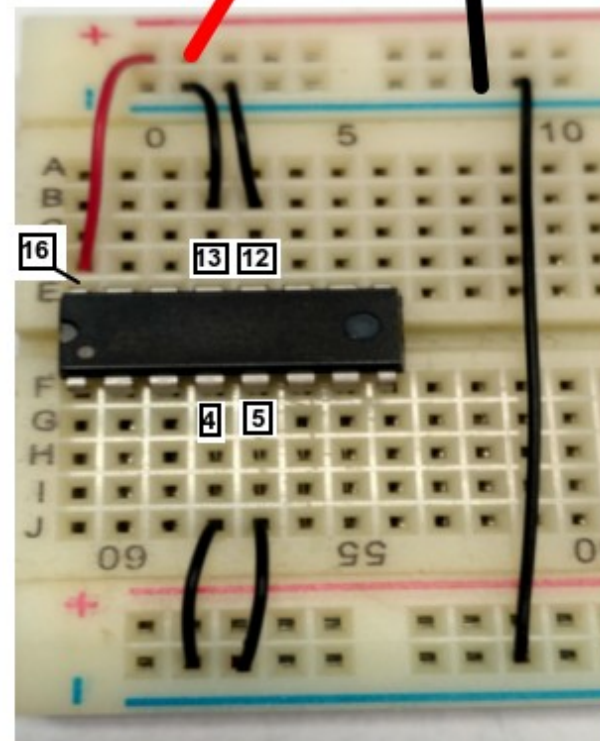
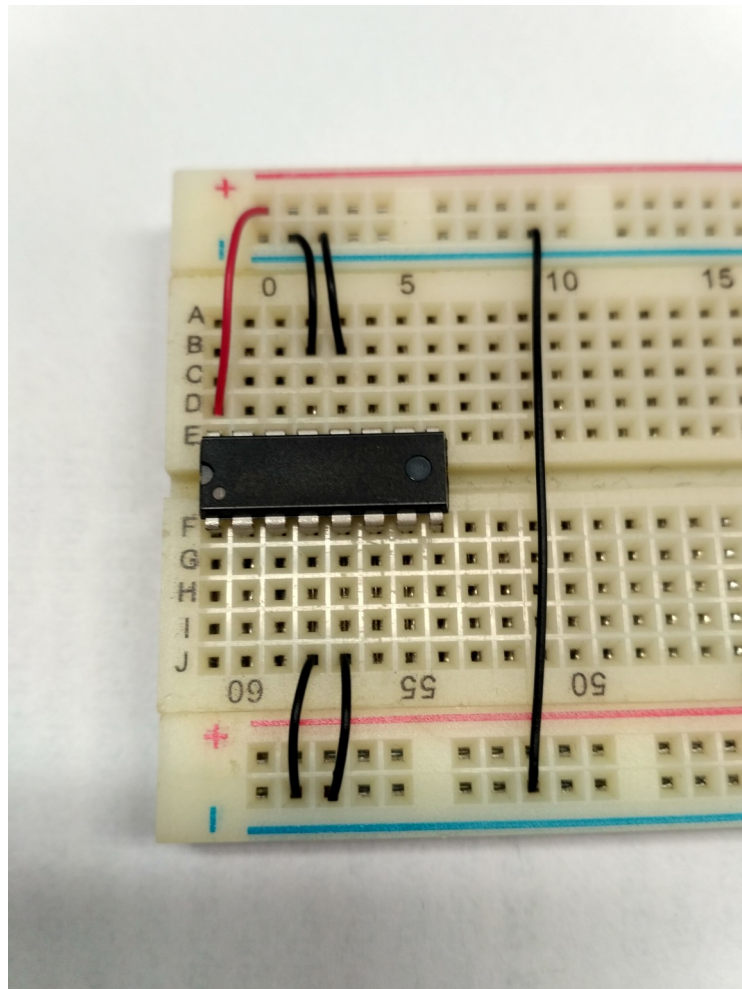


# Ligações - Energia



Pinos GND  
4, 5, 12, 13

Positivo +5V  
16



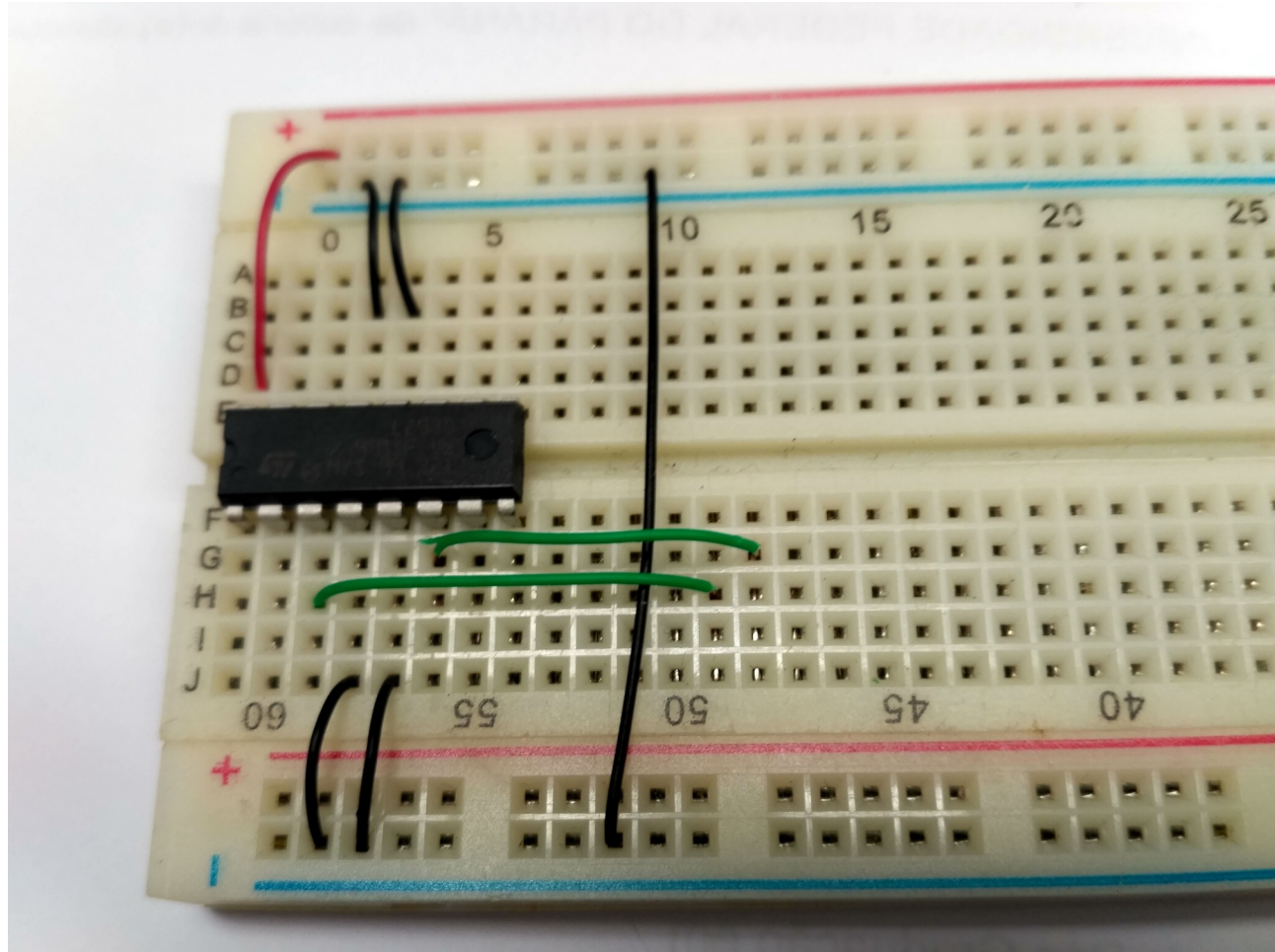
# Ligações – Motor ‘A’



Pinos GND  
4, 5, 12, 13

Positivo +5V  
16

Motor ‘A’  
3, 6





# Ligações – Motor ‘B’

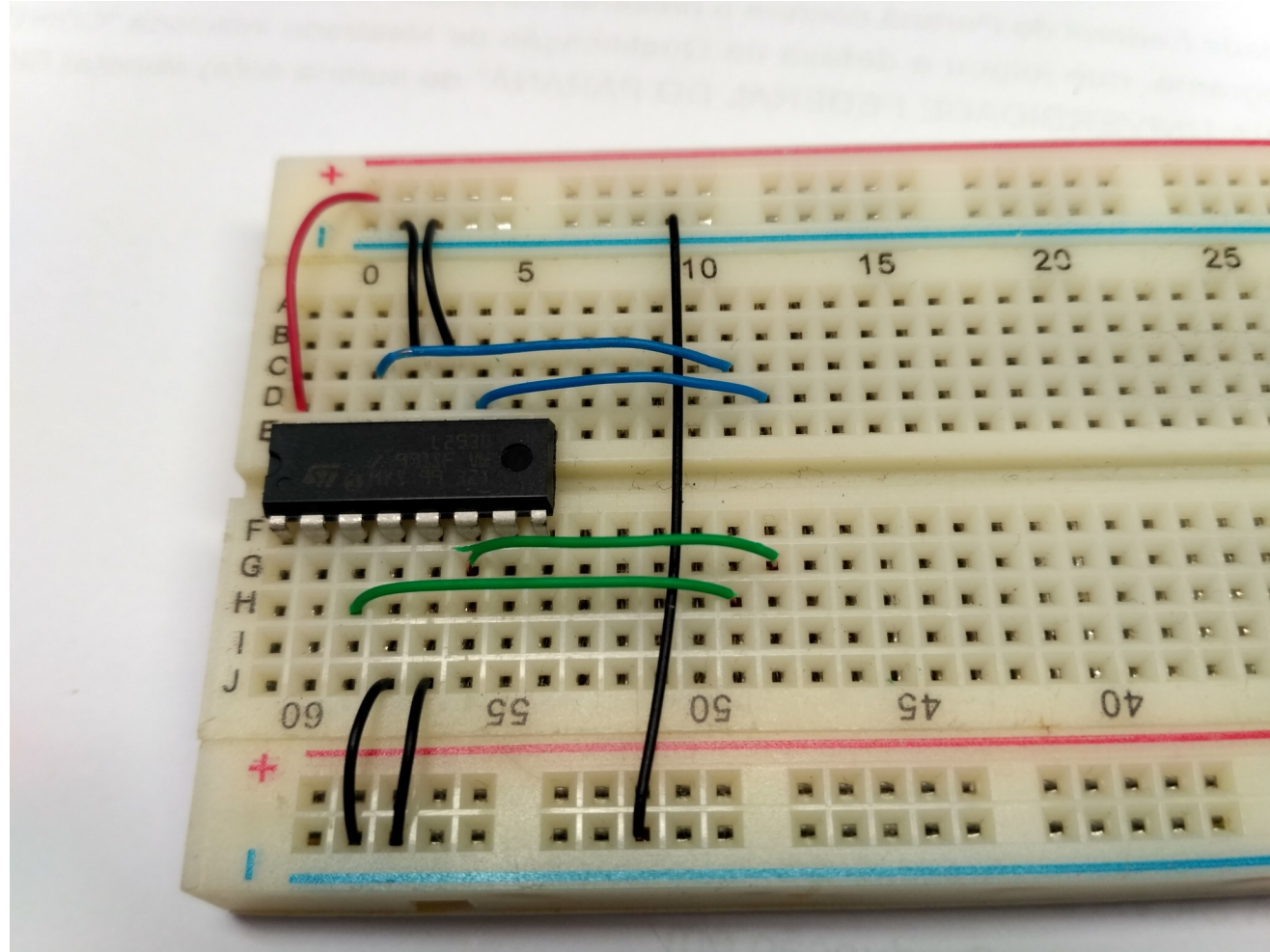


Pinos GND  
4, 5, 12, 13

Positivo +5V  
16

Motor ‘A’  
3, 6

Motor ‘B’  
11, 14



# Ligações - Motores

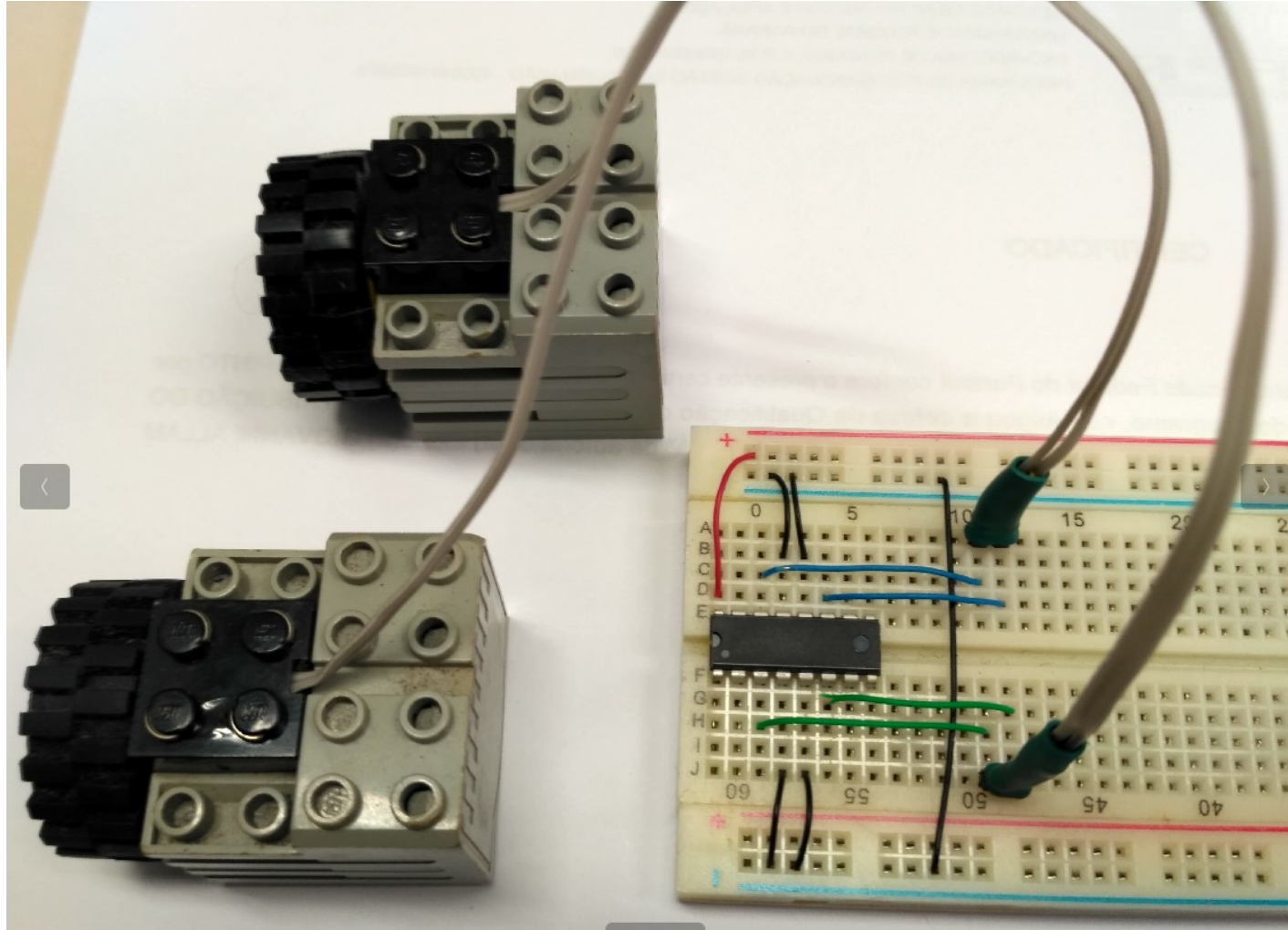


Pinos GND  
4, 5, 12, 13

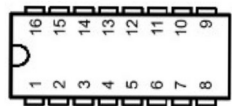
Positivo +5V  
16

Motor 'A'  
3, 6

Motor 'B'  
11, 14



# Ligações - Motores

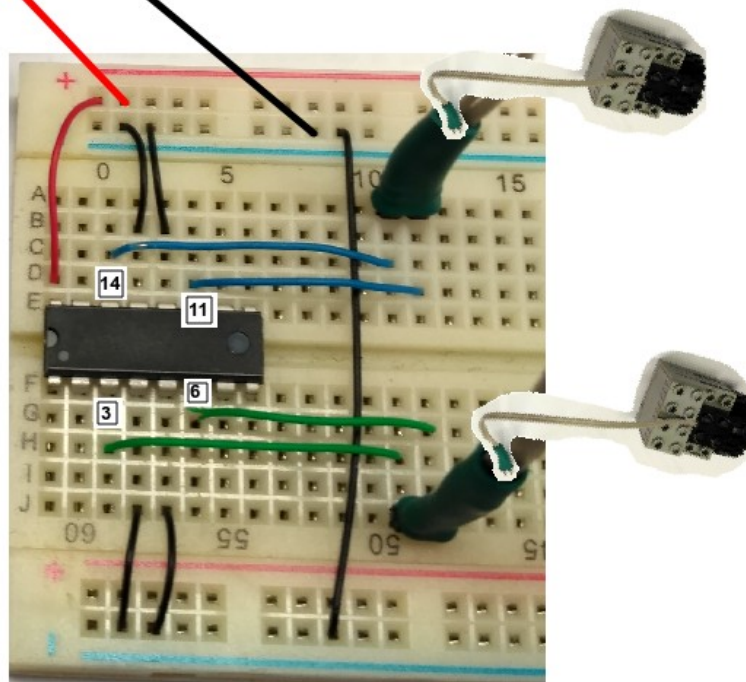


Pinos GND  
4, 5, 12, 13

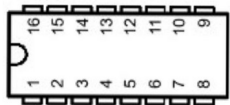
Positivo +5V  
16

Motor 'A'  
3, 6

Motor 'B'  
11, 14



# Ligações – energia para os motores

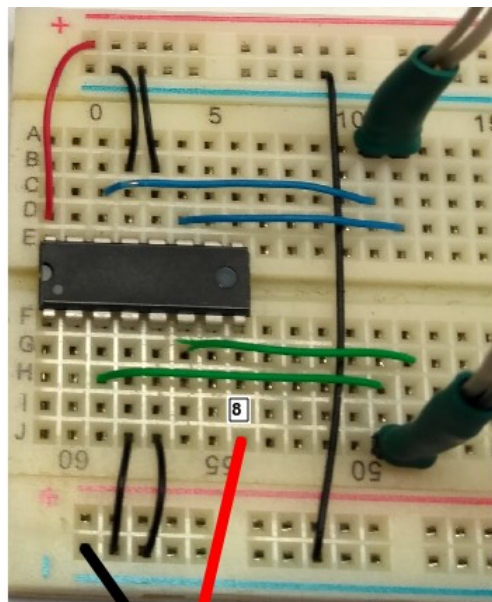


Pinos GND = 4, 5, 12, 13

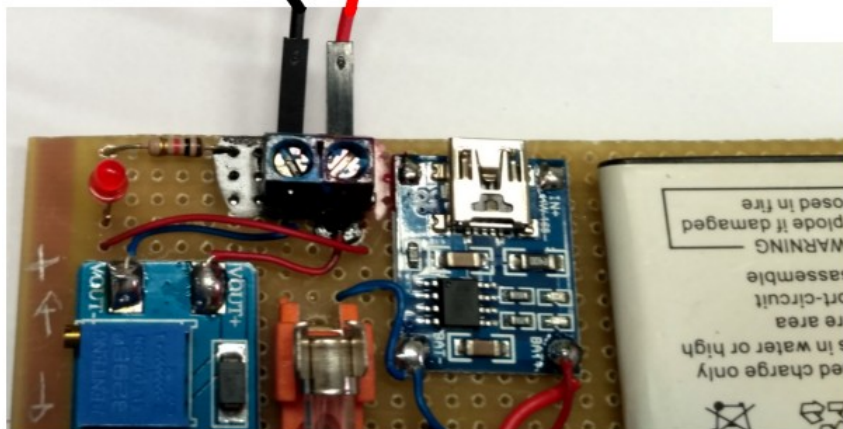
Positivo +5V = 16

Motor 'A' = 3, 6

Motor 'B' = 11, 14

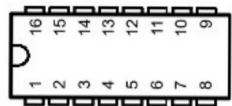


Energia para  
os motores  
GND no GND  
+9V no pino 8

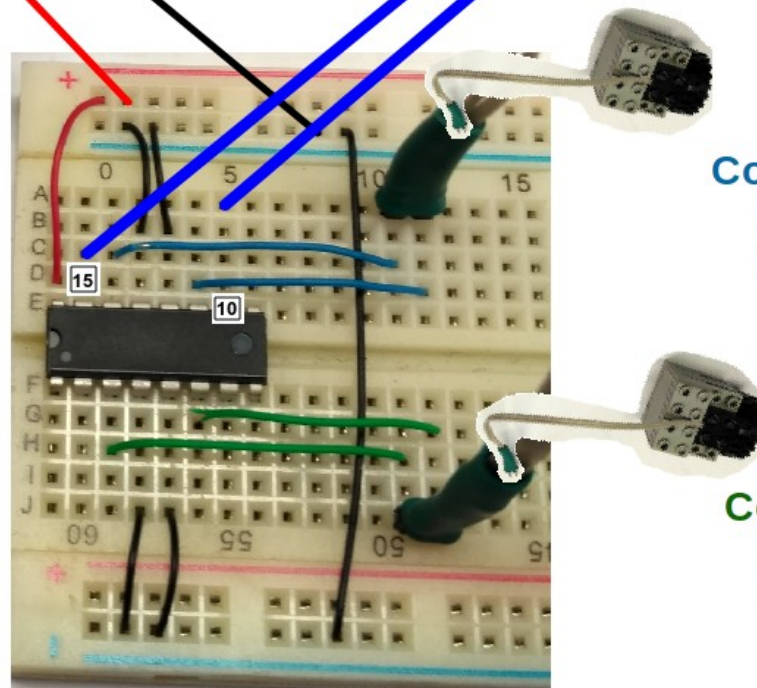
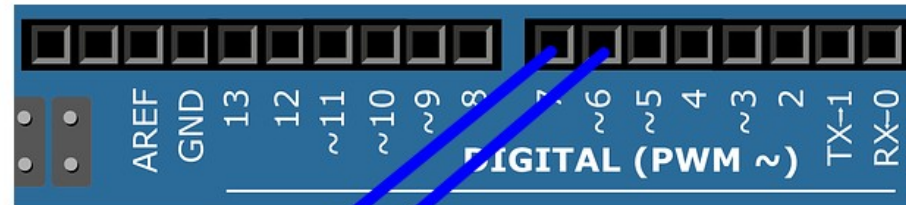




# Ligações – Controles de direção



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15



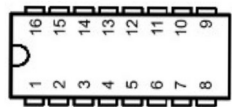
## Controle 'B'

10 do L293 = 6 Arduino  
15 do L293 = 7 Arduino

## Controle 'A'

2 do L293 = 9 Arduino  
7 do L293 = 8 Arduino

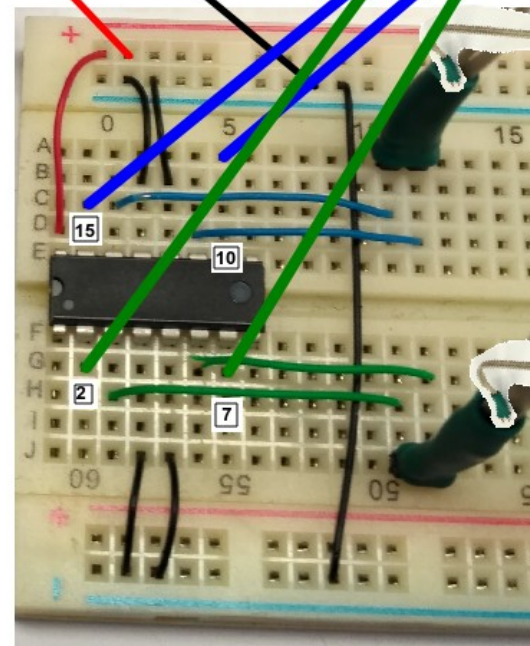
# Ligações – Controles de direção



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15

```
5 int EsquerdoA = 6; //Pino 10 do L293D
6 int EsquerdoB = 7; //Pino 15 do L293D
7 int DireitoA = 8; //Pino 7 do L293D
8 int DireitoB = 9; //Pino 2 do L293D
```

```
17 pinMode(EsquerdoA, OUTPUT);
18 pinMode(EsquerdoB, OUTPUT);
19 pinMode(DireitoA, OUTPUT);
20 pinMode(DireitoB, OUTPUT);
```



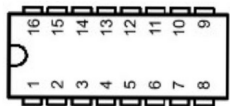
## Controle 'B'

10 do L293 = 6 Arduino  
15 do L293 = 7 Arduino

## Controle 'A'

2 do L293 = 9 Arduino  
7 do L293 = 8 Arduino

# Ligações – Controles de direção



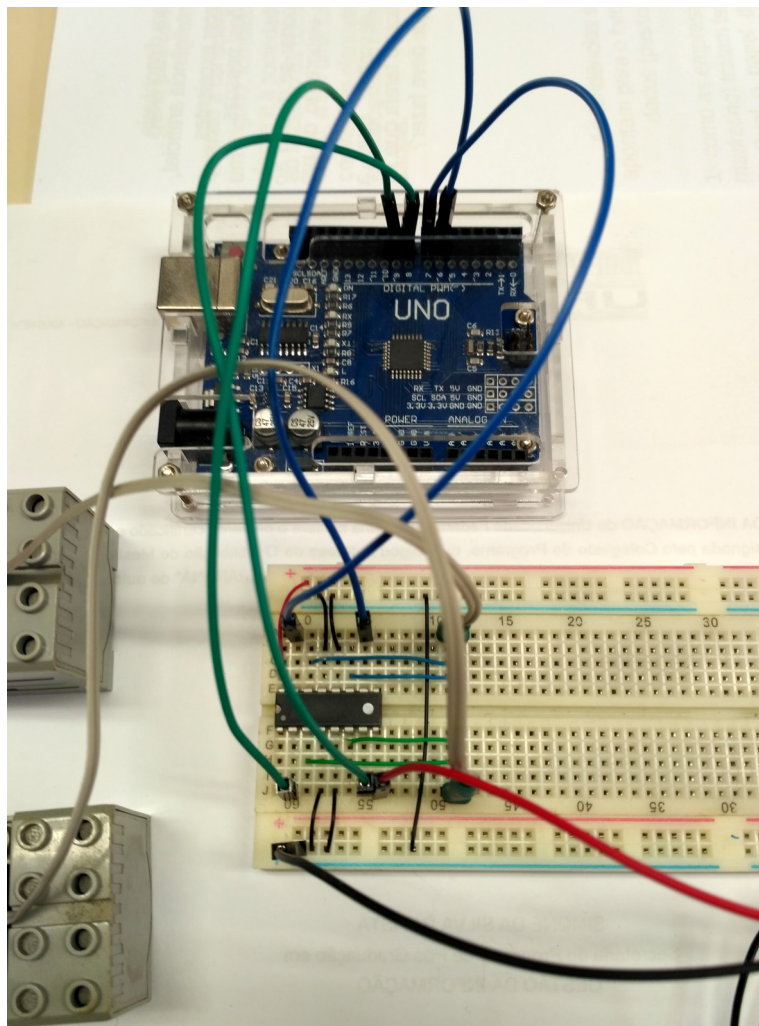
Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8

Controle 'A'

2, 7

Controle 'B'

10, 15



**Controle 'A'**

2 do L293 = 9 Arduino

7 do L293 = 8 Arduino

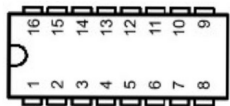
**Controle 'B'**

10 do L293 = 6 Arduino

15 do L293 = 7 Arduino



# Ligações – Controle de potência PWM



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15

## Potência 'A'

1

## Potência 'B'

9

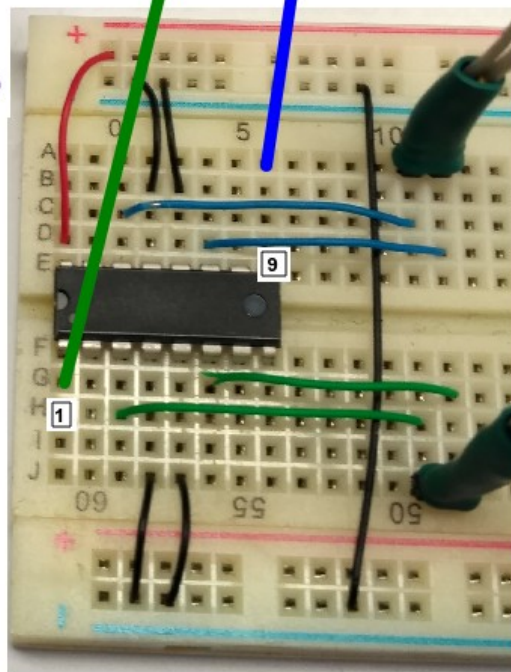


### Potência 'B'

9 do L293 = 3 Arduino

### Potência 'A'

1 do L293 = 5 Arduino

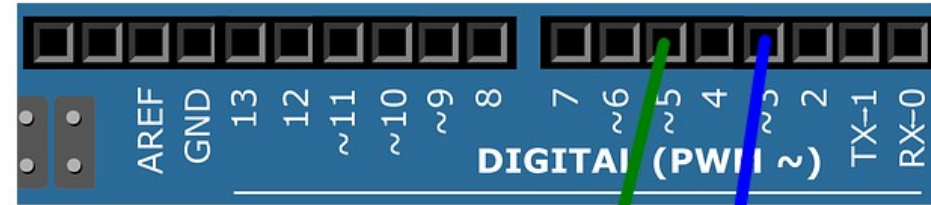




# Ligações – Controle de potência PWM



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot.= GND no GND  
+9V no pino 8  
Controle 'A' = 2,7  
Controle 'B' = 10, 15  
Potência 'A' = 1  
Potência 'B' = 9



**Potência 'B'**

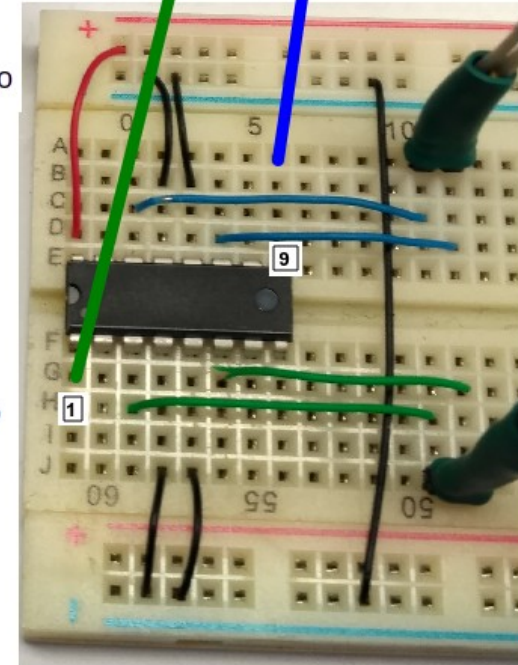
9 do L293 = 3 Arduino

**Potência 'A'**

1 do L293 = 5 Arduino

```
3 int ControleEsquerdo = 3; //Pino 9 do L293D
4 int ControleDireito = 5; //Pino 1 do L293D
9 int EnergiaEsquerdo = 100;
10 int EnergiaDireito = 100;
```

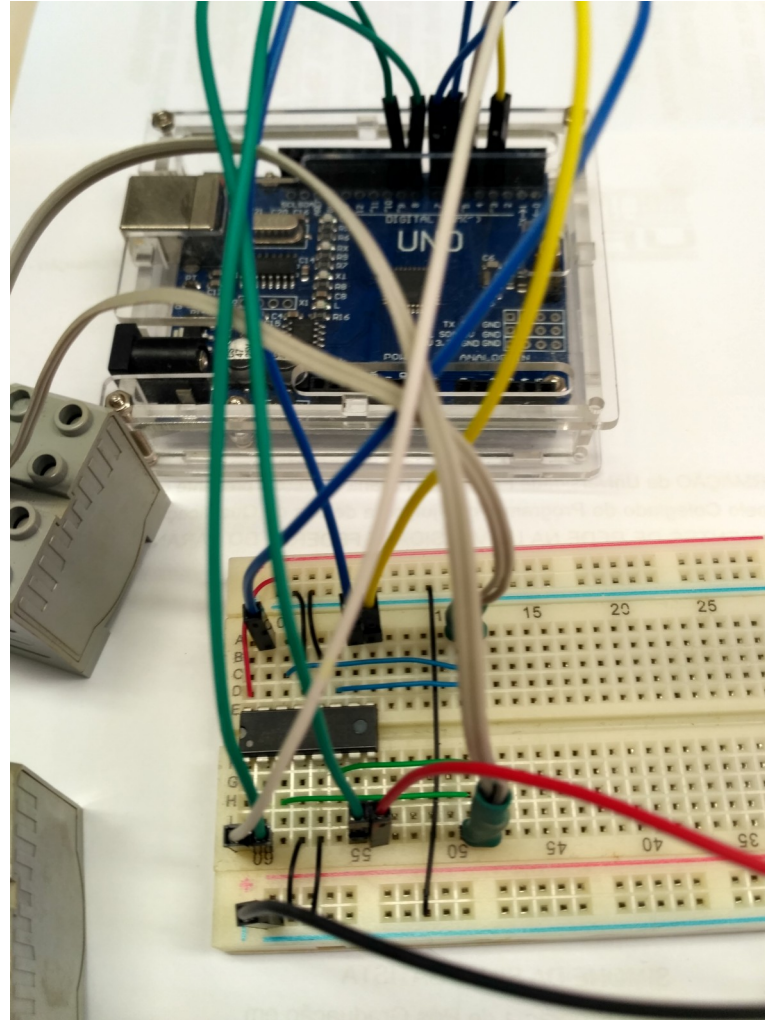
```
21 analogWrite(ControleEsquerdo, EnergiaEsquerdo);
22 analogWrite(ControleDireito, EnergiaDireito);
```



# Ligações – Controle de potência PWM



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15  
Potência 'A' = 1  
Potência 'B' = 9



## Potência 'A'

1 do L293 = 5 Arduino

## Potência 'B'

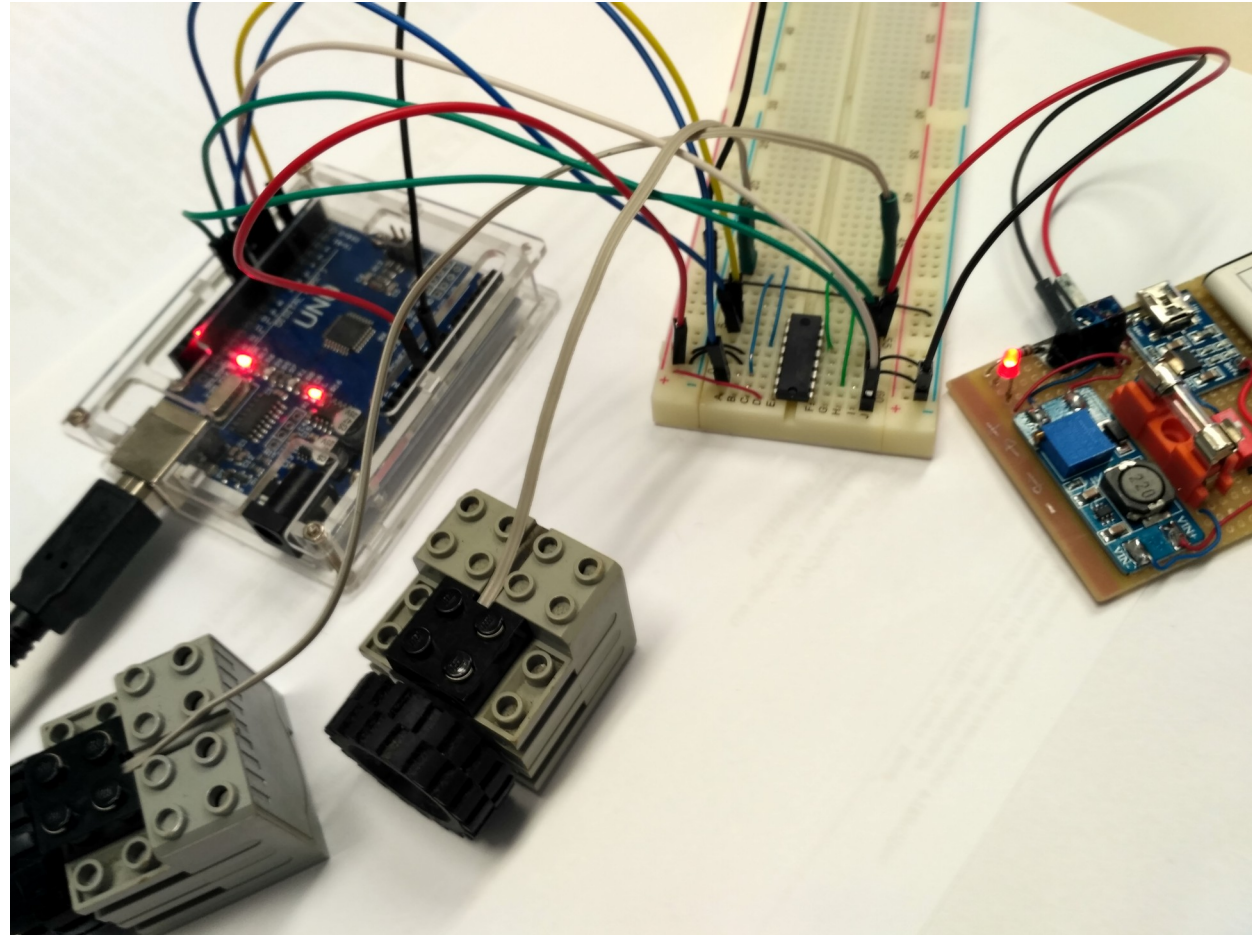
9 do L293 = 3 Arduino

(Pinos 3 e 5 são PWM)

# Ligações - completo



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15  
Potência 'A' = 1  
Potência 'B' = 9

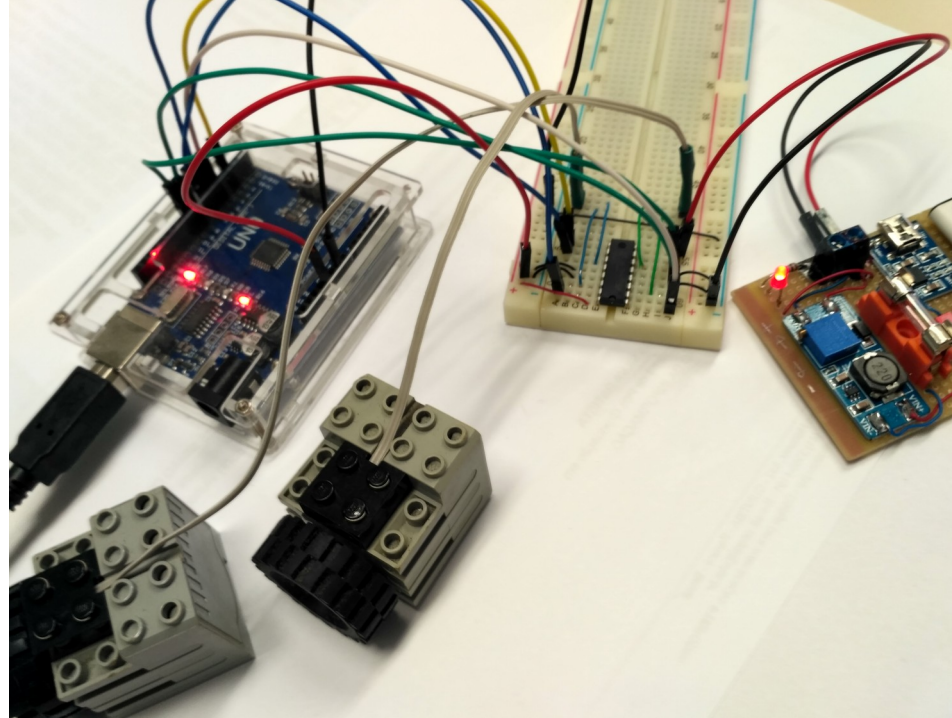




# Ligações – Visão geral



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15  
Potência 'A' = 1  
Potência 'B' = 9



## Controle 'A'

2 do L293 = 9 Arduino

7 do L293 = 8 Arduino

## Controle 'B'

10 do L293 = 6 Arduino

15 do L293 = 7 Arduino

## Potência 'A'

1 do L293 = 5 Arduino

## Potência 'B'

9 do L293 = 3 Arduino

(Pinos 3 e 5 são PWM)

# Ligações – Visão geral



Pinos GND = 4, 5, 12, 13  
Positivo +5V = 16  
Motor 'A' = 3, 6  
Motor 'B' = 11, 14  
Energia mot. = GND no GND  
+9V no pino 8  
Controle 'A' = 2, 7  
Controle 'B' = 10, 15  
Potência 'A' = 1  
Potência 'B' = 9



## Controle 'B'

10 do L293 = 6 Arduino

15 do L293 = 7 Arduino

## Potência 'B'

9 do L293 = 3 Arduino

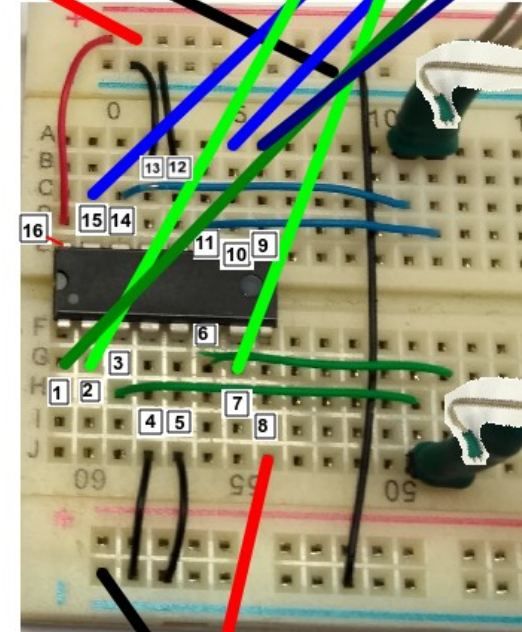
## Controle 'A'

2 do L293 = 9 Arduino

7 do L293 = 8 Arduino

## Potência 'A'

1 do L293 = 5 Arduino



Motor 'B' (9V)

Motor 'A' (9V)

A alimentação dos  
motores pode ter  
tensão diferente,  
no exemplo é de 9V



*Software*

# Declarações *hardware* x *software*

(se você usou outros pinos, não esqueça de ajustar...)

---

```
1 //Controle de motor usando o L293B
2
3 int ControleEsquerdo = 3; //Pino 9 do L293D
4 int ControleDireito = 5;  //Pino 1 do L293D
5 int EsquerdoA = 6; //Pino 10 do L293D
6 int EsquerdoB = 7; //Pino 15 do L293D
7 int DireitoA = 8; //Pino 7 do L293D
8 int DireitoB = 9; //Pino 2 do L293D
9 int EnergiaEsquerdo = 100;
10 int EnergiaDireito = 100;
```

# Inicialização

```
12 void setup( )
13 {
14     Serial.begin(9600);
15     pinMode(ControlEsquerdo, OUTPUT);
16     pinMode(ControlDireito, OUTPUT);
17     pinMode(EsquerdoA, OUTPUT);
18     pinMode(EsquerdoB, OUTPUT);
19     pinMode(DireitoA, OUTPUT);
20     pinMode(DireitoB, OUTPUT);
21     analogWrite(ControlEsquerdo, EnergiaEsquerdo);
22     analogWrite(ControlDireito, EnergiaDireito);
23 }
```



## Função - parar os motores

```
24 |  
25 void pararMotores() {  
26     digitalWrite(EsquerdoA, LOW);  
27     digitalWrite(EsquerdoB, LOW);  
28     digitalWrite(DireitoA, LOW);  
29     digitalWrite(DireitoB, LOW);  
30     delay(100);  
31 }
```

Finalidade – evitar transições bruscas na alimentação/ direção e restaurar potência ao original

# Funções - deslocamento

```
33 void emFrente(){
34     digitalWrite(EsquerdoA, HIGH);
35     digitalWrite(EsquerdoB, LOW);
36     digitalWrite(DireitoA, HIGH);
37     digitalWrite(DireitoB, LOW);
38 }
```

Note que o funcionamento HIGH/LOW LOW/HIGH inverte o sentido de rotação do motor

Se vai 'para frente' ou 'para trás' depende da montagem mecânica dos motores.

```
40 void reverso(){
41     digitalWrite(EsquerdoA, LOW);
42     digitalWrite(EsquerdoB, HIGH);
43     digitalWrite(DireitoA, LOW);
44     digitalWrite(DireitoB, HIGH);
45 }
```

# Funcionamento - *loop()*

```
47 void loop( )  
48 {  
49     if(Serial.available() > 0) {  
50         int val = Serial.parseInt();
```

Verifica se há algo na Serial e, se houver, lê número inteiro

# Funcionamento - *loop()*

Executa o teste e procedimentos em função do valor lido

```
51 | switch (val) {  
    52 |     case 0:  
    56 |     case 1:  
    61 |     case 2:  
    66 |     case 3:  
    81 |     case 4:
```

# Funcionamento - *loop()*

```
52 case 0:
53     Serial.println("Parado"); //esta linha pode ser comentada ou suprimida para uso com bluetooth
54     pararMotores();
55     break;
56 case 1:
57     Serial.println("Frente"); //esta linha pode ser comentada ou suprimida para uso com bluetooth
58     pararMotores();
59     emFrente();
60     break;
61 case 2:
62     Serial.println("Tras"); //esta linha pode ser comentada ou suprimida para uso com bluetooth
63     pararMotores();
64     reverso();
65     break;
```

# Funcionamento - *loop()*

```
66 case 3:
67     Serial.println("Curva à esquerda"); //esta linha pode ser comentada
68     pararMotores();
69     EnergiaEsquerdo -= 10;
70     if (EnergiaEsquerdo < 10){
71         EnergiaEsquerdo = 10;
72     }
73     analogWrite(ControleEsquerdo, EnergiaEsquerdo);
74     EnergiaDireito += 10;
75     if (EnergiaDireito > 250){
76         EnergiaDireito = 250;
77     }
78     analogWrite(ControleDireito, EnergiaDireito);
79     emFrente();
80     break;
```

Diminui a energia do motor Esquerdo, de forma que ele 'freie'

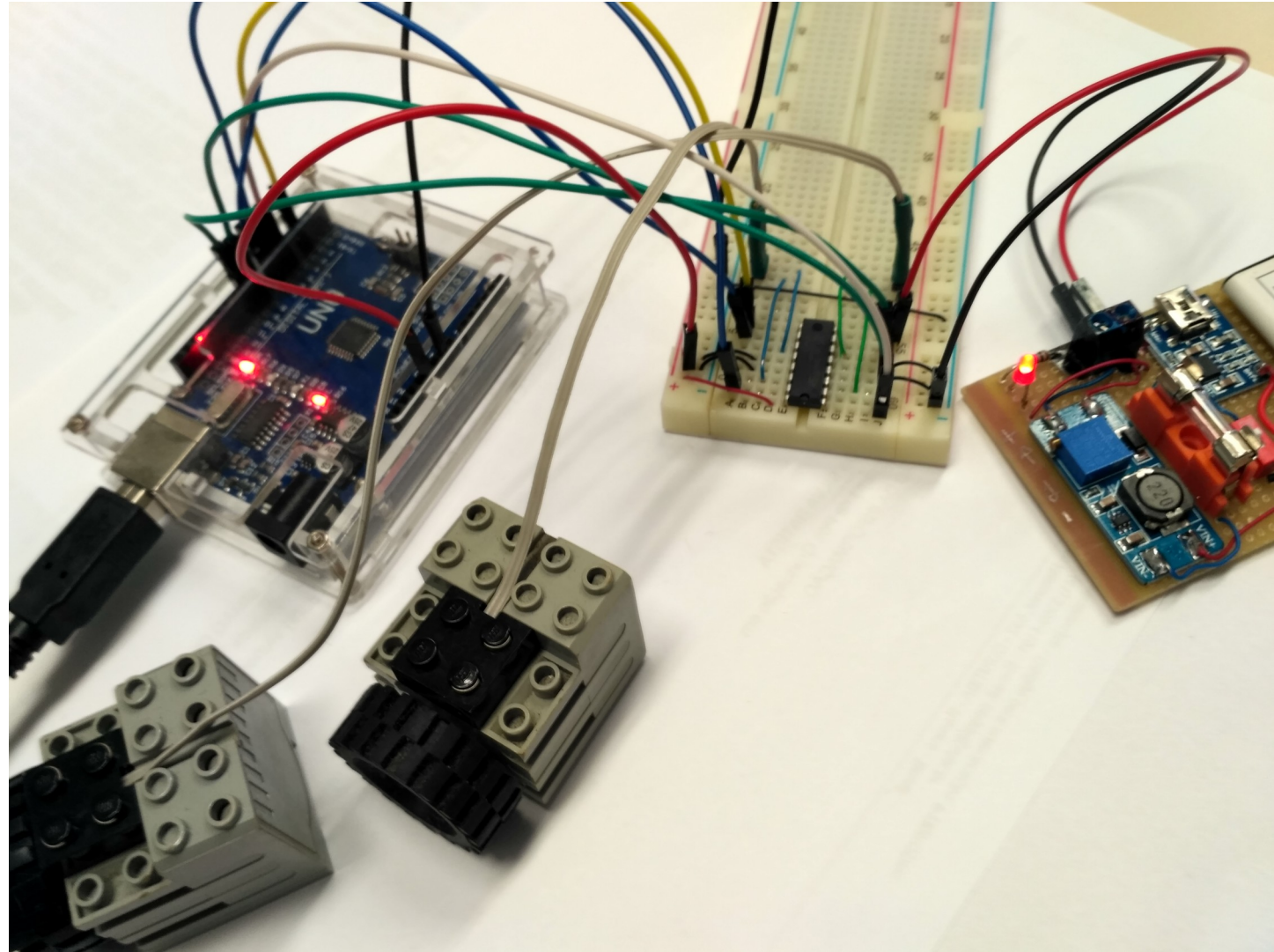
Aumenta a energia do motor Direito, de forma que ele 'acelere'

# Funcionamento - *loop()*

```
81 case 4:
82     Serial.println("Curva à direita"); //esta linha pode ser comentada
83     pararMotores();
84     EnergiaDireito -= 10;
85     if (EnergiaDireito < 10){
86         EnergiaDireito = 10;
87     }
88     analogWrite(ControleDireito, EnergiaDireito);
89     EnergiaEsquerdo += 10;
90     if (EnergiaEsquerdo > 250){
91         EnergiaEsquerdo = 250;
92     }
93     analogWrite(ControleEsquerdo, EnergiaEsquerdo);
94     emFrente();
95     break;
```

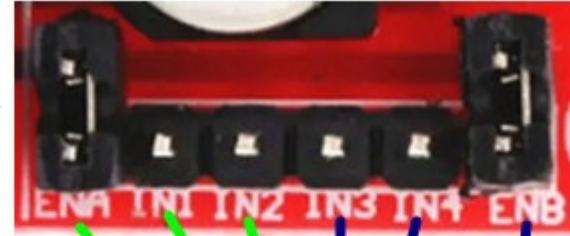
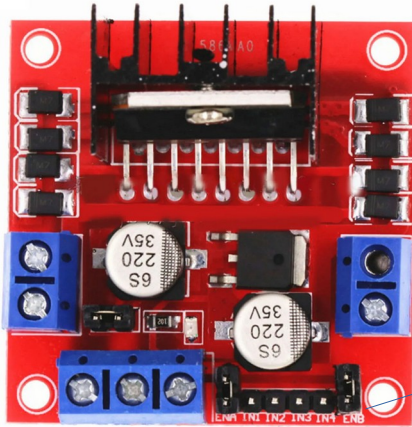
# Atividade 1

Montar o  
circuito com  
o L293 e o  
software  
proposto e  
testá-lo





# Outros controladores - L298N



‘Equivalência’

Habilitação do Motor 1 (ou PWM)	1	16	+5V para o L293
Motor 1/ controle A	2	15	Motor 2/ controle B
Motor 1/ saída 1	3	14	Motor 2/ saída 2
GND (0V)	4	13	GND (0V)
GND (0V)	5	12	GND (0V)
Motor 1/ saída 2	6	11	Motor 2/ saída 1
Motor 1/ controle B	7	10	Motor 2/ controle A
+V para os motores	8	9	Habilitação do Motor 2 (ou PWM)

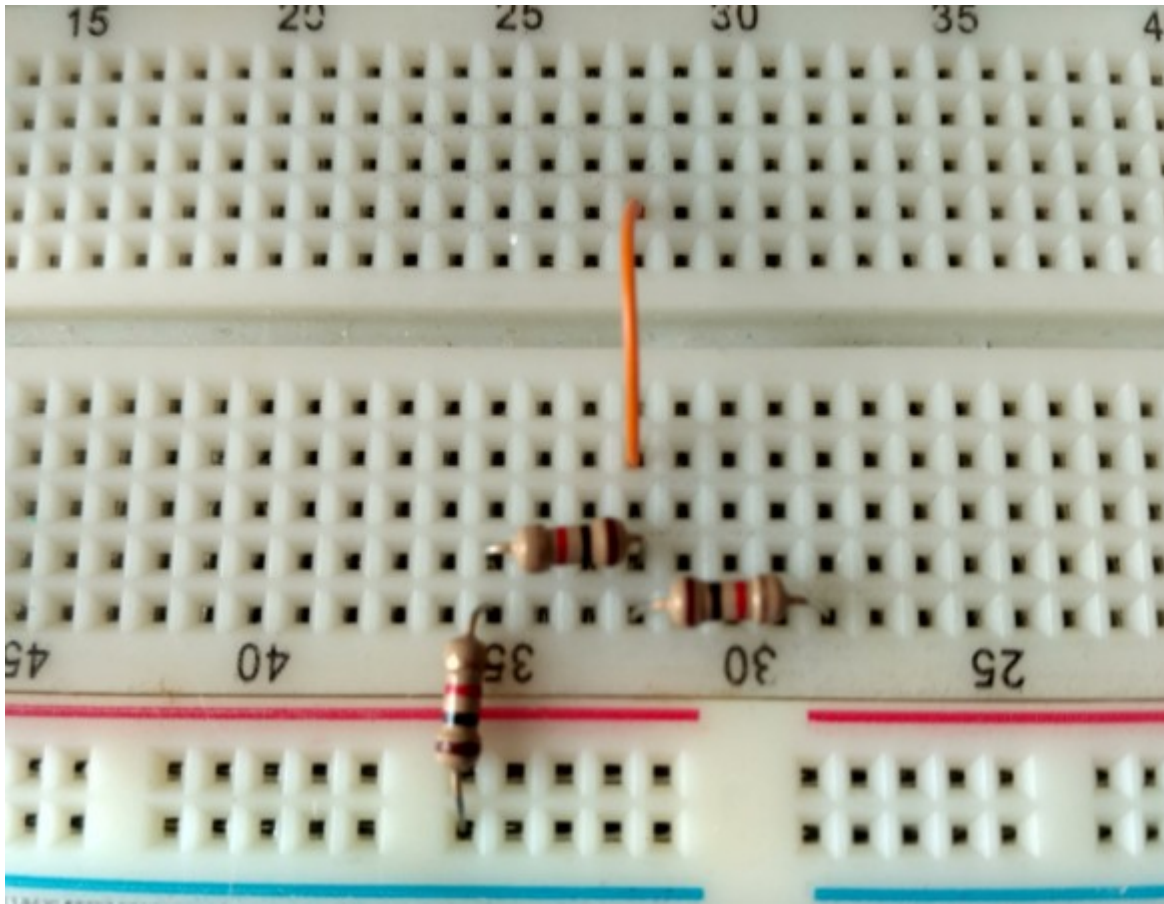
## Atividade 2

Inserir o  
módulo  
*bluetooth* e  
testá-lo

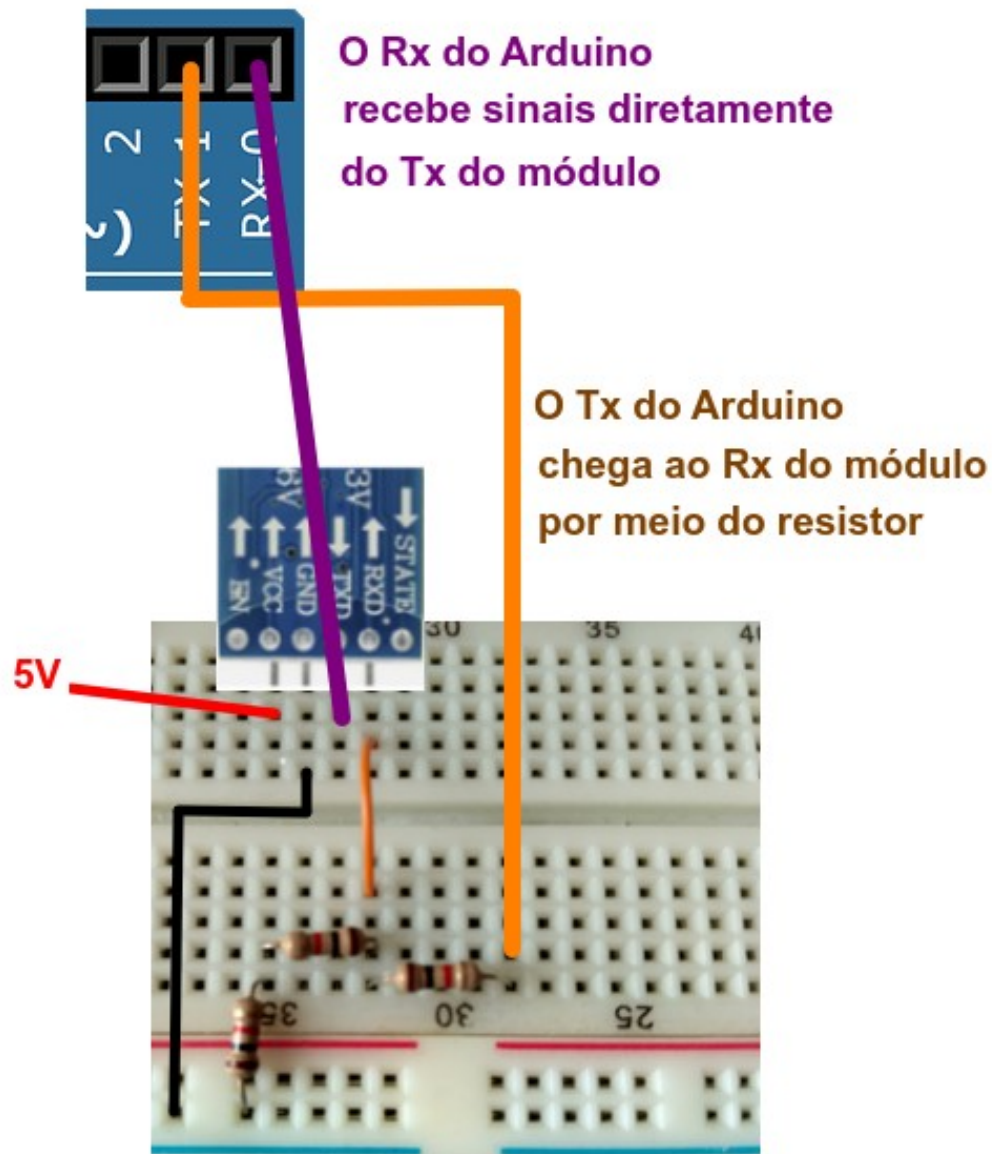




# Módulo *bluetooth* / Rx é para 3,3Volts



# Ligação do Módulo *bluetooth*



# Ligação no Arduino

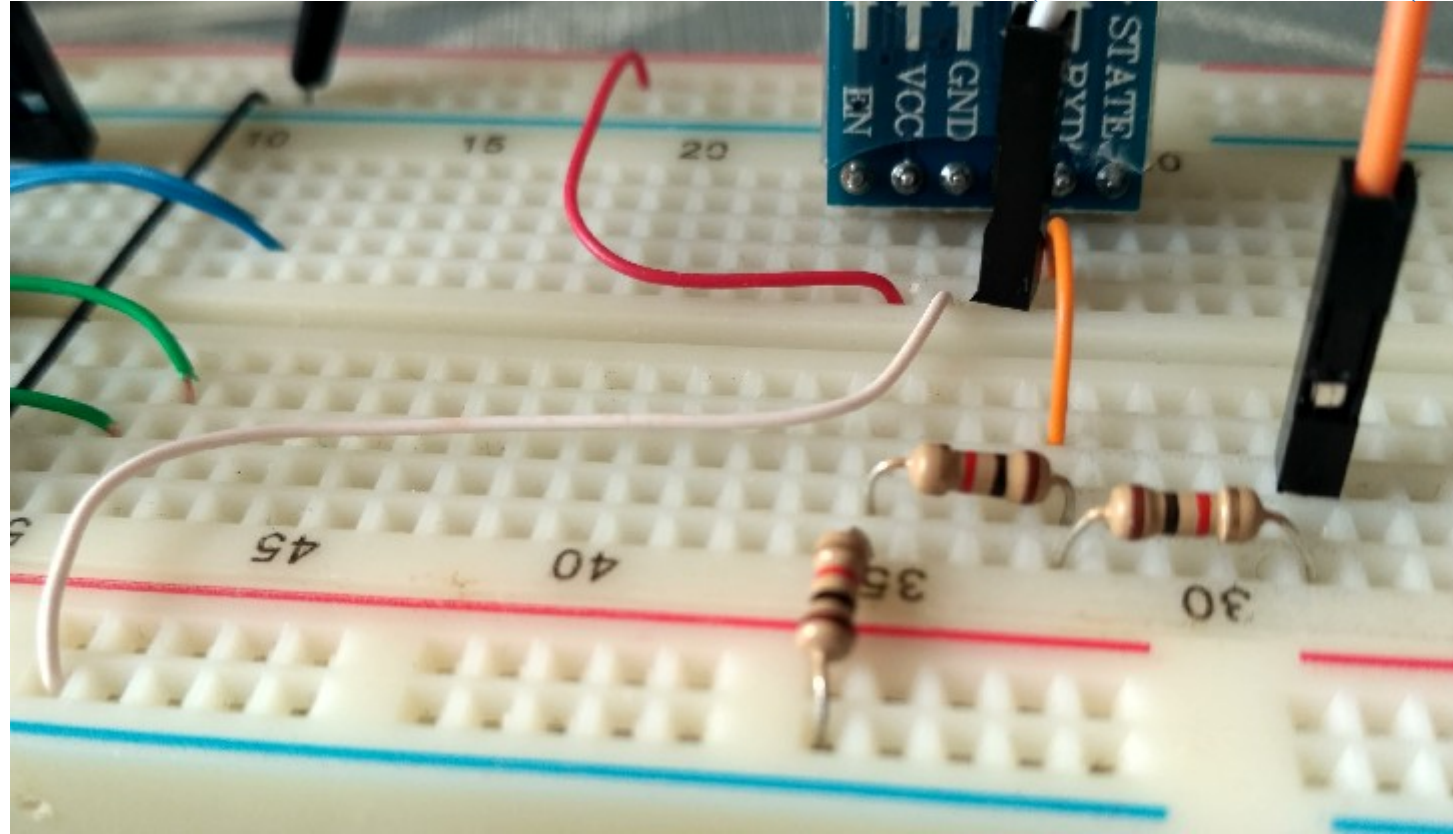
Resistor ligado  
ao divisor de  
tensão/ Rx do  
*bluetooth* no  
Tx do Arduino

e

Tx do módulo  
no Rx do  
Arduino

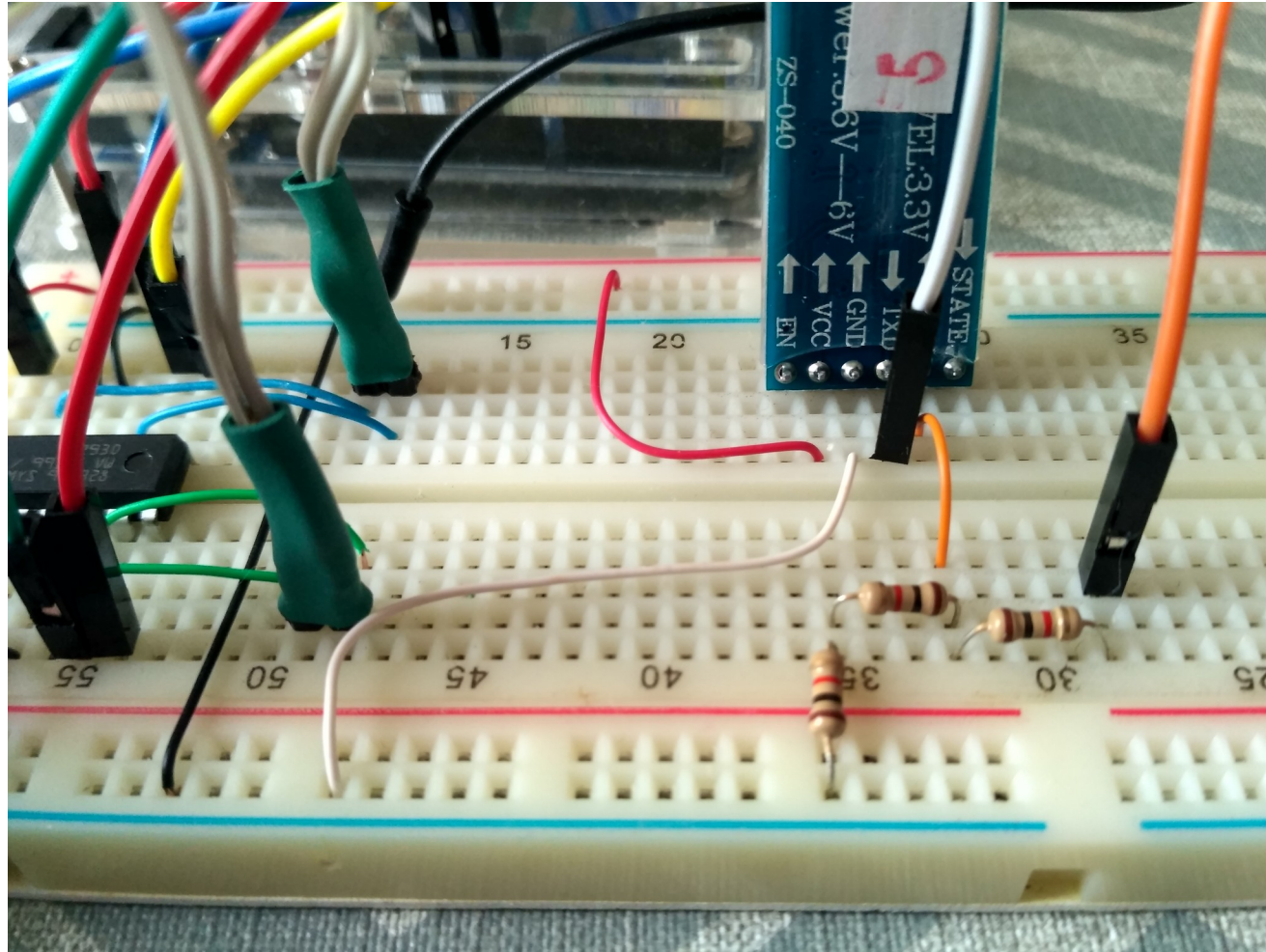
Ao Rx do  
Arduino

Ao Tx do  
Arduino



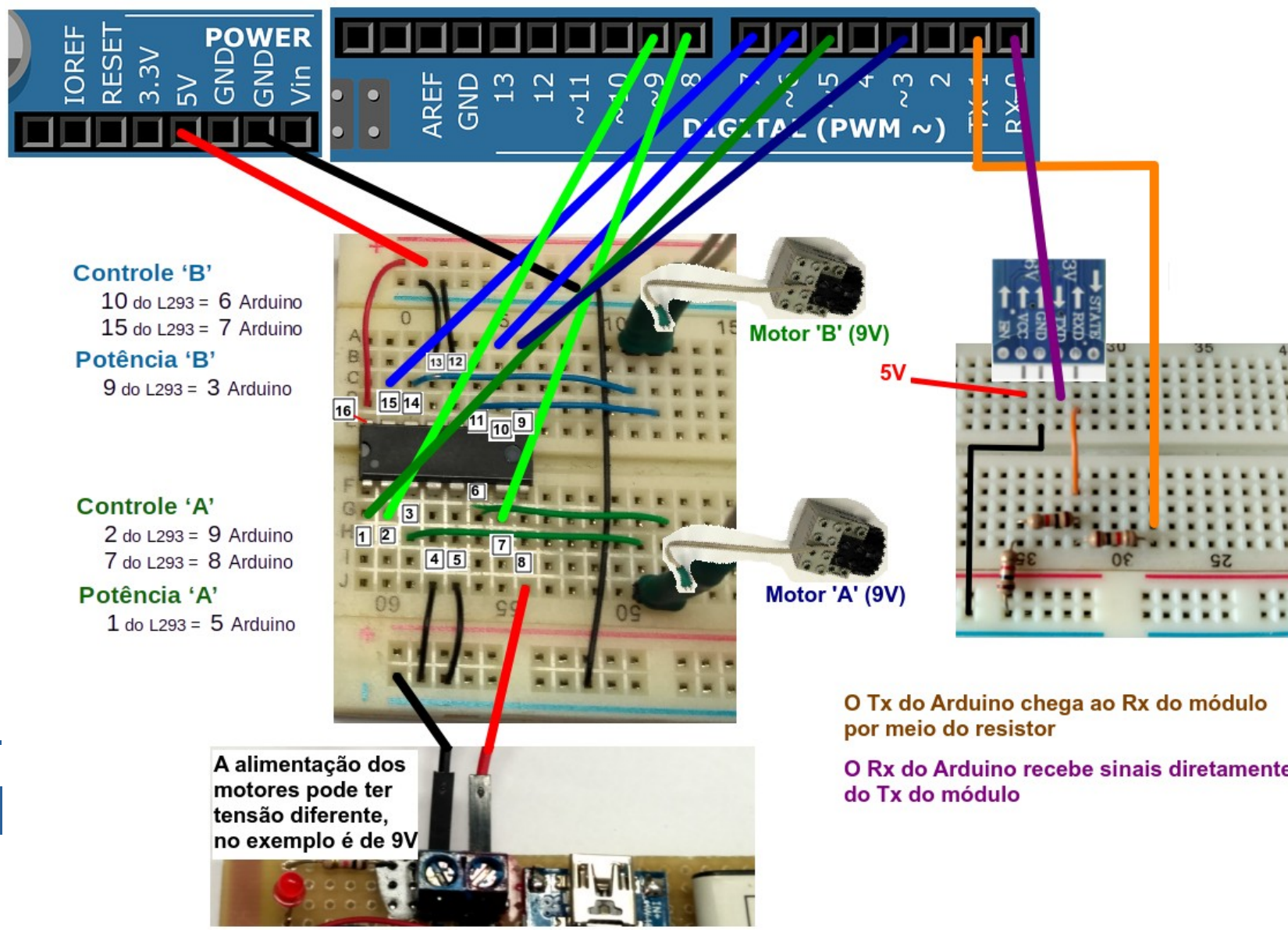


# Módulo *bluetooth* instalado

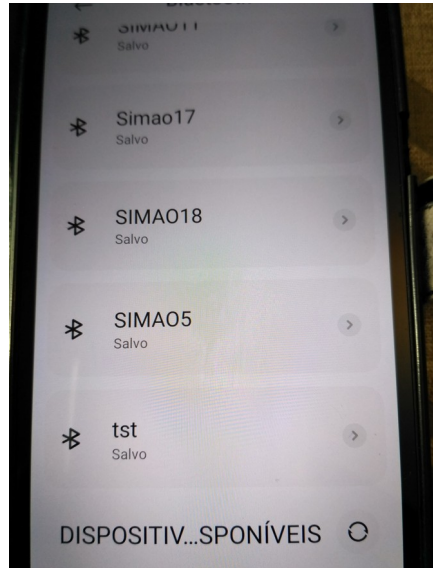




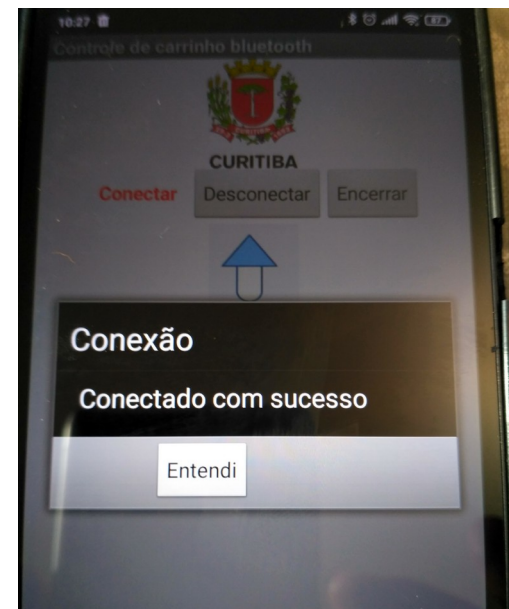
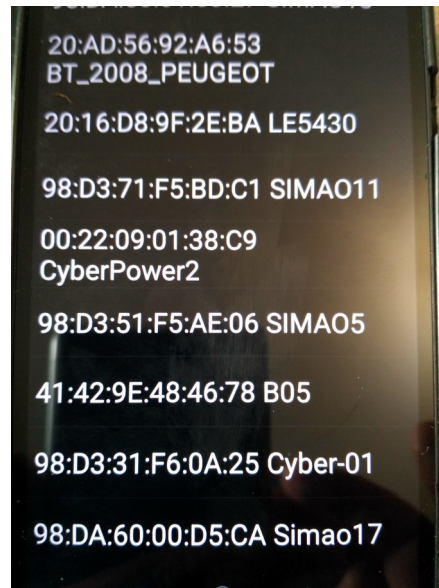
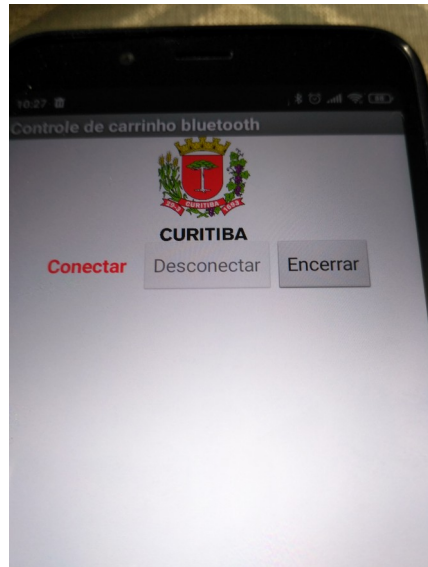
# Ligações – Visão geral



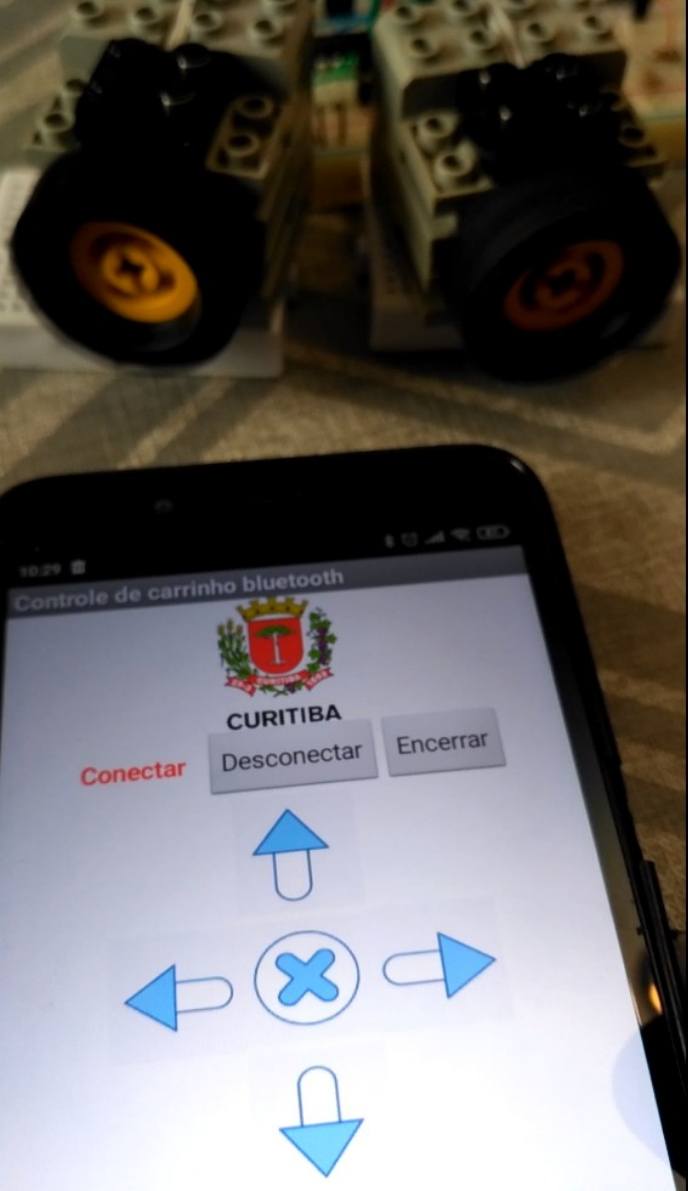
# Pareie o *bluetooth*



# Teste o App







Teste o App

